UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/998,363	11/29/2001	John Mathews	020431.1052	5533

5073	7590	08/27/2004		

BAKER BOTTS L.L.P.
2001 ROSS AVENUE
SUITE 600
DALLAS, TX 75201-2980

EXAMINER
STEELMAN, MARY J

ART UNIT	PAPER NUMBER
2122	

DATE MAILED: 08/27/2004

Please find below and/or attached an Office communication concerning this application or proceeding.

PTO-90C (Rev. 10/03)

	Application No.	Applicant(s)
Office Action Summary	09/998,363	MATHEWS, JOHN
	Examiner	Art Unit	
	Mary J. Steelman	2122	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE <u>3</u> MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.
- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on <u>*11/29/01.4/12/02,12/2/02*</u>.
2a)☐ This action is **FINAL**.   2b)☒ This action is non-final.
3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-48* is/are pending in the application.
    4a) Of the above claim(s) _____ is/are withdrawn from consideration.
5)☐ Claim(s) _____ is/are allowed.
6)☒ Claim(s) *1-48* is/are rejected.
7)☐ Claim(s) _____ is/are objected to.
8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☒ The specification is objected to by the Examiner.
10)☒ The drawing(s) filed on <u>*29 November 2001*</u> is/are: a)☒ accepted or b)☐ objected to by the Examiner.
    Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
    Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
    a)☐ All   b)☐ Some * c)☐ None of:
      1.☐ Certified copies of the priority documents have been received.
      2.☐ Certified copies of the priority documents have been received in Application No. _____.
      3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).
    * See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1)☒ Notice of References Cited (PTO-892)
2)☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3)☒ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date <u>*11/29/01*</u>.
4)☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5)☐ Notice of Informal Patent Application (PTO-152)
6)☐ Other: _____.

DETAILED ACTION

1.   Claims 1-48 are pending.

Specification

2.   Examiner objects to the length of the Abstract.

Content of Specification:

>   (j)   Abstract of the Disclosure: See MPEP § 608.01(f). A brief narrative of the
>   disclosure as a whole in a single paragraph of 150 words or less commencing on a
>   separate sheet following the claims. In an international application which has
>   entered the national stage (37 CFR 1.491(b)), the applicant need not submit an
>   abstract commencing on a separate sheet if an abstract was published with the
>   international application under PCT Article 21. The abstract that appears on the
>   cover page of the pamphlet published by the International Bureau (IB) of the
>   World Intellectual Property Organization (WIPO) is the abstract that will be used
>   by the USPTO. See MPEP § 1893.03(e).

Claim Rejections - 35 USC § 112

3.   The following is a quotation of the second paragraph of 35 U.S.C. 112:

>   The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the
>   subject matter which the applicant regards as his invention.

4.   Claims 5-7, 15, 21-23, 31, 36, and 46 are rejected under 35 U.S.C. 112, second

paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject

matter which applicant regards as the invention. Regarding claims 5, 15, 21, 31, 36, and 46, the

terms 'substantially' and 'essentially' are relative terms which renders the claim indefinite. The

terms "substantially" and "essentially" are not defined by the claim, the specification does not

provide a standard for ascertaining the requisite degree, and one of ordinary skill in the art would

not be reasonably apprised of the scope of the invention.

Claim Rejections - 35 USC § 103

5.   The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all

obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in
> section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are
> such that the subject matter as a whole would have been obvious at the time the invention was made to a person
> having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the
> manner in which the invention was made.

6.   Claims 1-22 are rejected under 35 U.S.C. 103(a) as being unpatentable over US Patent

5,600,789 to Parker et al., in view of US Patent 6,766,481 B2 to Estep et al.

Per claims 1, 17, and 32:

-system; method; software being embodied in computer-readable media;

(Parker: FIGs. 4 & 15, Col. 10, line 23, "GUI system under test:)

-centralized test queue operable to store a plurality of software GUI test instances to be executed

by a plurality of distributed test execution computers;

(Parker: Col. 1, lines 6-8, "...an apparatus and method for automatically testing Graphical User

Interfaces of computer systems", Col. 5, lines 16-62, "...tests and testing benchmarks are

portable between platforms...", col. 33, lines 41-58, "The test script of the present invention can

run on the same or on a different machine than the test driver or drivers being used at any one

time....test script....test executive...test driver, and application and GUI...The same test script is

thus able to drive different GUIs on 2 different machines...", col. 33, lines 65-66, "...a given test

executive can drive multiple targets simultaneously in a coordinated way...", FIG. 13 – plurality

of platforms, FIG. 15 -, plurality of testing.)

-a test server engine operable to, for each distributed test execution computer:

-receive a request for a software GUI test instance from a particular distributed test

execution computer in response to completion of a preceding software GUI test instance by the

particular distributed test execution computer;

(Parker: Col. 4, lines 44-46, "If the test was successful, the test executive continues to execute

the next step in the script...")

- retrieve a software GUI test instance from the test queue;

(Parker: Col. 4, lines 13-15, "The function of the test drives is to take the GUI specific

references from the test executive and perform the actual interface to the GUI objects.")

-communicate the retrieved software GUI test instance to the particular distributed test

execution computer for execution against a particular client-server combination using a testing

component supported by the particular distributed test execution computer, the testing

component operable to perform automated software GUI testing and to produce test results for

such testing for communication to the test server engine;

(Parker: FIG. 4, Col. 8, lines 37-40, "The test executive passes the GUI specific command to the

test driver. The test driver then performs the actual action of the GUI object specified in the test

script command.", col. 9, lines 58-61, "...command coded into test scripts will place text into the

same logical application object across different GUI platforms without requiring modification to

the command in the script", col. 15, lines 33-39, "The user input or series of inputs are

transmitted from the test driver to the GUI on interface. The test driver functions are specified in

the test script at a superclass, logical, generic level", col. 15, lines 60-62, "The test driver then

transmits the generated user input actions to the GUI thus simulating real user's input...performs

the requested action...", col. 33, lines 41-58.)

-receive a test result for the software GUI test instance from the particular distributed test

execution computer in response to execution of the software GUI test instance;

(Parker: Col. 30, lines 49-51, "When a script of group of scripts are executed, a single results

file is created which stores all information pertinent to the execution.")

-store the received test result for reporting to one or more users.

(Parker: Col. 31, lines 11-18, "A hypertext-like viewer could be used to view the results

file...one would see the scripts which executed and how many errors there were. If a script was

selected, one would see the test cases and how many error...")

Parker failed to supply specific details regarding "each distributed test execution

computer comprising a client platform and coupled to one or more server platforms, the client

platforms and server platforms collectively providing a plurality of client-server combinations

against which the software GUI test instances may be executed". However Estep disclosed a

software test system. Estep disclosed consideration to (col. 4, lines 7-8) "various component

areas that make up the architecture of the desired software system." Col. 4, lines 59-67,

"...determines the component areas required for building the target software system. More

particularly...network management system...needed to design, develop and implement..."

Therefore, it would have been obvious, to one of ordinary skill in the art, at the time of the invention to include information regarding various network platform combinations in a GUI software test environment, as that allows for more fully testing of the software.

Per claim 2, 18, and 33:

-at least one distributed test execution computer operates at a location geographically remote from the other distributed test execution computers and from the test server.

(Parker: FIG. 15 – distributed test execution.)

Per claims 3, 19, and 34:

Parker fails to address:

-the testing component is a commercial off-the-shelf product.

However, Estep, disclosed a software testing system. At col. 3, lines 36-39, Estep disclosed, "gathers the appropriate commercial off-the-shelf (COTS) software products for testing…"

Therefore, it would have been obvious, to one of ordinary skill in the art, at the time of the invention, to have modified Parkers GUI testing to include commercial off the shelf software as it is well known in the art, and an effective reuse of software, thereby saving time and money in software development.

Per claims 4, 20, and 35:

-each software GUI test instance is an instance of a software GUI test written using a test

scripting language and can be executed using any of the distributed test execution computers, a

software GUI test instance being executed using the particular distributed test execution

computer from which the request initiating retrieval of the software GUI test instance from the

test queue was received.

(Parker: Col. 34, lines 4-6, "Since the test tool has test drivers for all of the popular GUIs, a

given test script can drive not only multiple targets simultaneously, but multiple heterogeneous

targets.", Col. 4, lines 8-9, "The test executive executes the test script.")

Per claims 5, 21, and 36:

Parker generated test results, but failed to disclose limitations related to web page displays.

However, Estep disclosed:

-the test server engine is further operable to generate a test results web page comprising test

results for a plurality of software GUI test instances, including the test result for the most

recently executed software GU'I test instance, substantially immediately upon receiving the test

result from the particular distributed test execution computer on which the most recently

executed software GUI test instance was executed;

-the system further comprises a web server operable to communicate the test results web page for

display on a user system to provide substantially real-time test results reporting.

Col. 3, lines 66- col. 4, line 2, "All finalized reports are posted on a global computer

network,e.g., the World Wide Web…wherein it is accessible by any interested customer or

consumer."

Therefore, it would have been obvious, to one of ordinary skill in the art, at the time of

the invention, to have modified Parker's test executive to produce test results on a web page, as

this is a well known manner to publish information, effectively and inexpensively providing real

time details to customers.

Per claims 6, 22 and 37:

-each software GUI test instance is an instance of a software GUI test;

(Parker: Col. 4, lines 13-20, "The function of the test drives is to take the GUI specific

references from the test executive and perform the actual interface to the GUI objects. At the

time of execution of the test script, the test executive and the test driver take the references to the

logical objects contained in the script and translate them...to identify, manipulate and query the

actual objects under test...)

-the test results web page comprises consolidated test results for a particular client platform, the

consolidated test results indicating test results for each software GUI test for each client-server

combination involving the particular client platform.

(Estep provided details regarding posting results to a web page (col. 3, lines 66- col. 4).

Therefore, it would have been obvious, to one of ordinary skill in the art, at the time of

the invention, to have modified Parker's test executive to produce test results on a web page, as

this is a well known manner to publish information, effectively and inexpensively providing real

time details to customers.

Per claims 7, 23, and 38:

Parker disclosed an invention whereby a test script is provided to execute desired features of a

GUI. Parker failed to disclose test selections via a results web page:

-the test server engine is further operable to receive a user request to execute an instance of a

particular software GUI test and to insert the requested software GUI test instance into the test

queue according to the user request, the user request being input by selecting the particular

software GUI test using the test results web page.

However, Estep provided details regarding posting results to a web page (col. 3, lines 66-

col. 4).

Therefore, it would have been obvious, to one of ordinary skill in the art, at the time of

the invention, to have modified Parker's test executive to produce test results on a web page

allowing a developer to further choose test scripts from the web page, as a web page is a well

known manner of displaying information, effectively and inexpensively allowing interactions.

Per claims 8, 24, and 39:

-at least some GUI test instances in the test queue have associated priorities, the test server

engine operable to retrieve the GUI test instances from the test queue for execution according to

their associated priorities.

(Parker: Col. 4, lines 15-20, "...test executive and the test driver take the references...and

translate them into a form...to identify, manipulate and query (retrieve according to

priorities)...)

Per claims 9, 25, and 40:

-the test queue comprises a first queue containing higher priority software GUI test instances and a second queue containing lower priority software GUI test instances, the test server engine operable to retrieve higher priority software GUI test instances from the first queue for execution during a first part of a testing period and retrieve lower priority software GUI test instances from the second queue for execution during a second part of the testing period.

(Parker: Col. 4, lines 15-20, "...test executive and the test driver take the references...and translate them into a form...to identify, manipulate and query (retrieve according to priorities)...)

Parker does not specify "a first queue...a second queue...higher / lower priorities..." However, he does disclose that the executive and drivers may identify, manipulate and query objects under test for the purposes of execution.

Therefore it would have been obvious, to one of ordinary skill in the art, at the time of the invention to have modified Parker's invention to include variations of priorities during multiple test executions because he disclosed manipulations of the objects being tested, thus producing more varied tests.

Per claims 10, 26, and 41:

-the test server engine is operable to re-communicate instances of a software GUI test for execution against all client-server combinations, according to a rule, in response to receiving one or more test results for the software GUI test indicating failure.

(Parker: Col. 4, lines 15-20, "...test executive and the test driver take the references...and translate them into a form...to identify, manipulate and query (retrieve according to priorities)...)

Parker does not specify "re-communicating instances of a test...in response to receiving results..." However, he does disclose that the executive and drivers may identify, manipulate and query objects under test for the purposes of execution. He also disclosed, col. 4, lines 46-50, "If the test was not successful, the test executive can invoke an exception handler to decide how to proceed with the test..."

Therefore it would have been obvious, to one of ordinary skill in the art, at the time of the invention to have modified Parker's invention to include "to re-communicate instances of a software GUI test for execution against all client-server combinations, according to a rule, in response to receiving one or more test results for the software GUI test indicating failure", because he disclosed manipulations of the objects being tested, and error handling options, thus re-communicating instances for execution re-test would provide added assurance of results, allowing for parameter modifications.

Per claims 11, 27, and 42:
-the test server engine is operable to detect when the number of software GUI test instances in the test queue is below a predefined threshold and, in response, to automatically add software GUI test instances to the test queue.

(Parker: Col. 4, lines 1-5, "The first component is a test script which is written in a high level

programming language and contains the user events to be simulated, and the control and data

structures necessary (maintain predefined threshold) to validate the GUIs...")

Per claims 12, 28 and 43:

-a client controller associated with each distributed test execution computer and operable to

automatically install a current software GUI build at each distributed test execution computer at

one or more appropriate times during a testing period.

(Parker: FIG. 11, #692 & #682, Col. 26, line 27, "Changes to the user interface...(install current

build)...", col. 26, lines 31-34, "The testing methodology provided by the present invention

allows the principles of software engineering to be applied to the testing of applications as well

as to the development of such programs.")

Per claims 13, 29, and 44:

-a client controller associated with each distributed test execution computer and operable

automatically reboot each distributed test execution computer according to predetermined

schedule.

(Parker: Col. 3, lines 60-61, "The present invention is directed at testing both new and revised

computer application programs that use a Graphical User interface(GUI)...", col. 4, line 4,

"...control and data structures necessary to validate the GUIs...", col. 16, lines 42-43, "System

functions allow the test executive to perform operating system functions (reboot)...")

Per claims 14, 30, and 45:

-a client controller associated with each distributed test execution computer and operable to

establish communication with the test server engine when the distributed test execution computer

boots up.

(Parker: Col. 4, lines 4-5, "...control and data structures necessary to validate the GUIs...")

Per claims 15, 31, and 46:

-each test execution computer operates essentially as an automated test execution robot,

repeatedly requesting, receiving, executing, and returning test results for software GUI test

instances, automatically and without human intervention, for an extended time period.

(Parker: Col. 27, lines 15-29, "...test script is driven by data contained in a functional

specification repository...", col. 27, lines 58-61, "On of the strengths of the present invention is

the ability to use and re-use object data collected during a baseline execution..." Automatically

repeating according to a functional specification, re-using data collected.)

Per claim 16:

-further comprising the distributed test execution computers.

(Parker: FIG. 15.)

Per claim 47:

A system for distributed automated software GUI testing, comprising:

(Parker: Col. 5, line 17, '...test system...')

-means for maintaining a centralized test queue operable to store a plurality of software GUI test

instances to be executed by a plurality of distributed test execution computers, each distributed

test execution computer comprising a client platform and coupled to one or more server

platforms, the client platforms and server platforms collectively providing a plurality of client-

server combinations against which the software GUI test instances may be executed;

-means for receiving a request for a software GUI test instance from each particular distributed

test execution computer in response to completion of a preceding software GUI test instance by

the distributed test execution computer;

-means for retrieving a software GUI test instance from the test queue in response to the request

from the particular distributed test execution computer;

-means for communicating the retrieved software GIJI test instance to the particular distributed

test execution computer for execution against a particular client-server combination using a

testing component supported by the particular distributed test execution computer, the testing

component operable to perform automated software GUI testing and to produce test results for

such testing;

-means for receiving a test result for the software GUI test instance from the particular

distributed test execution computer in response to execution of the software GUI test instance;

-means for storing the received test result for reporting to one or more users.

(See limitations as addressed in claim 1 above.)

Per claim 48:

A system for distributed automated software graphical user interface (GUI) testing, comprising:

-a centralized test queue operable to store a plurality of software GUI test instances to be executed by a plurality of distributed test execution computers, each distributed test execution computer comprising a client platform and coupled to one or more server platforms, the client platforms and server platforms collectively providing a plurality of client-server combinations against which the software GUI test instances may be executed, each software GUI test instance is an instance of a software GUI test written using a test scripting language and executable using any of the distributed test execution computers;

-a test server engine operable to, for each distributed test execution computer:

-receive a request for a software GUI test instance from the particular distributed test execution computer in response to completion of a preceding software GUI test instance by the particular distributed test execution computer;

-retrieve a software GUI test instance from the test queue;

-communicate the retrieved software GUI test instance to the particular distributed test execution computer for execution against a particular client-server combination using a testing component supported by the particular distributed test execution computer, the testing component operable to perform automated software GUI testing and to produce test results for such testing for communication to the test server engine, a software GUI test being executed using the particular distributed test execution computer from which the request initiating retrieval of the software GUI test from the test queue was received;

-receive a test result for the software GUI test instance from the particular distributed test execution computer in response to execution of the software GUI test instance;

-store the received test result for reporting to one or more users in a test results database;

-generate a test results web page comprising the test results for the plurality of software GUI test instances.

-each distributed test execution computer operating essentially as an automated test execution robot, repeatedly requesting, receiving, executing, and returning results for software GUI test instances automatically without human intervention for extended periods of time;

-a web server operable to:

-access the test results database to obtain test results for a plurality of software GUI test instances;

-communicate the test results web page for display on a user system to provide substantially real-time test results reporting.

(See limitations as addressed in claims 1, 4, 5, and 15 above.)

Conclusion

7.   The prior art made of record and not relied upon is considered pertinent to applicant's disclosure.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Mary Steelman, whose telephone number is (703) 305-4564. The examiner can normally be reached Monday through Thursday, from 7:00 AM to 5:30 PM   If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Tuan Dam can be reached on (703) 305-4552. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Information regarding the status of an application may be obtained from the Patent

Application Information Retrieval (PAIR) system. Status information for published applications

may be obtained from either Private PAIR or Public PAIR. Status information for unpublished

applications is available through Private PAIR only. For more information about the PAIR

system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR

system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Mary Steelman

08/20/2004

ANTONY NGUYEN-BA
PRIMARY EXAMINER